

Lessons Learned from IT Projects

August 31, 2006

Legal (Contract) Items

1. Document the commitments and accountabilities of all involved parties, as well as project ownership and decision-making. Although the contracts must focus on the chains of command, roles and responsibilities between the ownership departments and the primary vendors, they should also include an overview of other parties associated with each project. Accordingly they should describe the reporting relationships, duties, and lines of authority for oversight and regulatory bodies, supporting and enabling organizations, and other involved parties, within and outside the ownership department. Product and/or service requirements and the project plans should provide the foundation for defining commitments and accountabilities.
2. Provide for a software escrow agreement for developed code or modified COTS packages, including agent/custodian, activation conditions, updates (version control), instructions/documentation, testing of loading procedures, tools and utilities (licenses to use), and authentication of code (latest version, complete environment and validity). The software must be escrowed within the geographic boundaries of the state.
3. Provide for perpetual licenses of purchased and developed software, including responsibilities of the state and vendors, rights to change code by the state and vendors (or third parties), warranties, and ownership of enhancements (especially those that can be separated from the original vendor code).
4. Delineate ownership of system components and data. This includes the rights of the state to convert, sell or transfer the system, any of its components, and/or data to other public or private entities.
5. Ensure vendors are licensed to do business in North Carolina. This can be verified through the Office of the Secretary of State.
6. Take actions to limit the fiscal damage to the state if the vendors fail to perform as required, including performance bonds, letters of credit, etc. Become knowledgeable of the limits and activation requirements of these vehicles.
7. Identify and delineate the long-term maintenance and support of software and other products or facilities, including responsibilities of the state and vendors, pricing/fees, performance commitments, and alternatives and flexibility of the state in the event of non-performance by the vendors.

Lessons Learned from IT Projects

8. Delineate project milestones or reference points that are measurable and verifiable. These form the foundation for status reporting, performance measurement, and vendor payments for the project.
9. Incorporate penalties and incentives by specifying benchmarks, commitments and performance measures (schedule, quality, costs, etc.) for vendors.
10. Determine and delineate vendor quality assurance, project management, technical, and documentation requirements and evaluations. Verify the existence and adequacy of vendor practices, processes and standards and the compliance of these processes with state requirements. This includes on-site (vendor locations) reviews and evaluations; the specifying of expected processes, standards and measures (e.g., IEEE, SEI, ISO, state technical architecture, state quality assurance practices, etc.); and the enumeration of penalties and rewards. Access by state personnel (ownership department and staff of regulatory and oversight bodies) and third party quality assurance personnel to vendor facilities and documents must be allowed on an uncontested basis throughout the duration of project.
11. Incorporate a reasonable payment schedule that balances the cash flow requirements of vendors with the fiduciary responsibility of the state to pay only for the delivery of acceptable products and/or services per schedule, quality and deliverable milestones and commitments and other appropriate performance measures.
12. Specify and describe change management practices and procedures, including identification and approval processes; approaches for effecting design, coding and other changes; timetables for implementing changes; version control procedures; testing and validation practices; documentation commitments; cost/fee structures; and payment approvals and schedules.
13. Specify and describe known changes and modifications to purchased software, including design, coding, testing and documentation practices; timetables for implementation; version control; and costs.
14. Specify vendors' practices and procedures for project planning and tracking and project management and reporting, as well as proposed systems development life cycle methodologies (SDLC). This includes the types, frequency and contents of reports that must be submitted by vendors. As a minimum, the formats and specifications for reports to the project or agency should be defined.
15. Provide assurance of the legality of third party software (and other third party products) incorporated in the system by identifying the software/products and providing proof of license. Determine responsibilities for one-time and ongoing license and maintenance payments. Delineate the one-time and ongoing responsibilities, procedures and costs associated with maintaining system integrity when third party components are updated or changed by their vendors.

Lessons Learned from IT Projects

16. Specify the physical geographic location of vendors' staff for performing the project to facilitate appropriate management and control of these personnel resources. Try to have vendor staff at the same location as the state staff.
17. Specify the physical location of software during development, as well as the procedures for controlling access and updates. This is necessary to maintain version control of the software and its integrity.
18. Ensure all provisions for the prime (lead) vendors also apply to the subcontractors, especially those regarding processes and procedures for project management and reporting, quality assurance, software development practices and processes, technical reviews, location of staffs.
19. Document privacy and security requirements. The requirements for ensuring the privacy of the state's public and private citizens and the protection of assets (information, hardware, software, etc.) must be identified and specified.
20. Enforce provisions of the contract. Unfavorable deviations in performance must be anticipated, identified, addressed and resolved in a timely and effective manner, especially when aggressive timetables are involved.
21. Consider the provision for a formal arbitration process. If appropriate, ensure the contract provides for and permits arbitration (possibly binding) as a problem resolution vehicle. Arbitration allows the project to continue while potential contract problems are resolved.
22. Develop and implement a security plan based on a realistic assessment of product vulnerabilities and security requirements of the Statewide Technical Architecture. Establish appropriate administrative, technical, and physical safeguards to protect the state and the software from unauthorized usage.
23. Develop a centralized file for contract task orders and other contract documentation. A comprehensive change management process must be formally defined, implemented, and monitored throughout the system life cycle. Centralize both paper and electronic copies.
24. Ensure that the vendor has a formal, documented configuration management process in place for software and documentation deliverables. The vendor configuration management plan should be verified and validated (tested) along with the agency configuration management plan. Configuration management must include change management, version control and review and approval procedures, and it must include all deliverables including software, hardware, and documentation.
25. Develop a formal vendor response format to an RFP and ensure that all proposals adhere to the defined guidelines for form and format. Structured responses allow for comparison and evaluation on an equal basis and facilitates the analysis of vendor

Lessons Learned from IT Projects

responses and best value procurement. Vendors may and should offer innovative and creative solutions, but the response format must be consistent.

26. Ensure that user acceptance criteria are clearly defined in the contract. Acceptance testing must be rigorous and comprehensive. Both the vendor and the agency must understand and agree to the deliverable acceptance criteria.
27. Business functional requirements and design documents for system modification efforts must be extremely detailed and are best developed with the cooperation and participation of the vendor. Once the contract is signed, ensure that the vendor participates in the requirements and design processes. Time to build these documents should be included in the contract.
28. Select the right vendor. Ensure that the contract is a “partnership” with the appropriate scoring model (measurement system) in place.
29. Ensure that the vendor rules for responding to RFPs are explained to the vendors so the agencies do not have to justify eliminating a vendor when not completely and appropriately responding to mandatory requirements.
30. Do not use a fixed-price contract for very large software development projects when customer requirements are not known in advance. Use time and materials contract for large software development projects.
31. Ensure that fixed-priced contract has mechanisms to enforce vendor performance, such as award-free pools, penalties or liquidated damages.
32. Require a particular software development capability from the vendor, such as CMMI-3.
33. Ensure that the Agency and Statewide procurement office use common practices. Ownership of procurement responsibility must be established before publication of any procurement documentation.
34. Remember, a low response rate by potential bidders, or bids that are not within acceptable ranges, or bids with a wide disparity of proposed prices should raise a “red flag” that requirements were not clearly defined, properly communicated, or are unrealistic.
35. Develop an upfront scoring model for vendor that allows for a methodical evaluation of the proposals. Participation by various agency users is also important.

Lessons Learned from IT Projects

Project Management, Risk Analysis and Quality Assurance Items

1. Ensure project managers and staff have adequate skills, training and experience in required disciplines, including technology, business/program, administration and project management. Skills and experience must be commensurate with the complexity, size and scope of the project and the assigned roles and responsibilities of individual team members.
2. Ensure that the department responsible for the project recognizes ownership of the effort and is responsible and accountable for the results. The department can not abdicate its obligation for the ultimate delivery of the project on time, on budget, and with the expected capabilities and benefits.
3. Establish a management steering committee for each project. Incorporate other advisory bodies as necessary for soliciting advice and knowledge from affected individuals, groups and organizations. Steering committee members should include representation from stakeholder groups, subject matter experts (SME), and the project sponsor's organization. Get everyone involved in the process.
4. Document the chain of command for the projects and the roles, commitments, responsibilities and accountabilities of all involved parties. Affected entities include the ownership department, oversight and regulatory bodies, supporting and enabling organizations, and steering committees, and advisory bodies. Use memos or documents of understanding, service level agreements, etc. to ensure all parties have a clear agreement of commitments and accountabilities. Ensure that no parties commit to more than they can deliver.
5. Evaluate the project's organization structure and ensure the appropriate checks and balances are employed, especially for an independent project manager. The state must control the project; therefore, the senior project manager should be a state employee or an outside contractor independent from the primary vendor.
6. Define and implement risk management policies and procedures to ensure major risks are identified, scrutinized and minimized from the beginning of the project. This includes the review and evaluation of technical, schedule, financial, management and strategic risks. A true understanding of the size and complexity of the project risk profile must be obtained, and the actions necessary for mitigating significant risk factors must be accomplished. Risks must be planned for, monitored, and controlled throughout the development effort.
7. Include communications in risk management evaluations. For statewide projects, especially those involving communications with local governments or remote state offices, recognize the impacts of adverse weather situations (e.g., thunderstorms, hurricanes, tornadoes, etc.) and potential technical problems on the reliability of transmissions. Remote and central data base updates,

Lessons Learned from IT Projects

redundancy, and backup and restoration procedures must accommodate these situations.

8. Develop a comprehensive, relevant and realistic project plan before beginning work. Spend time to develop a realistic plan. Where multiple entities/detailed plans are involved, a master plan must be used to integrate tasks and to plan, monitor and report overall project activities and status. The plan must include user involvement; participation of oversight, partnership and enabling organizations (e.g., ITS, Attorney General's office, Department of Administration (P&C), etc.); provisions for business and systems disaster recovery; and internal and outside quality assurance reviews (financial, management and technical). Breaking the project down into discrete and manageable components (Work Breakdown Structure) facilitates the development of the plan. Cognizant and responsible department management and oversight and regulatory bodies must approve the plan, and it must be kept current and valid. Always have a good project plan. If you do not plan well – plan often!!!
9. Ensure that planning phases are limited in scope to cover only the implementation intended over a reasonable period of time. Projects should be defined in small increments to ensure delivery rather than just planning.
10. Ensure that standard project management methods and practices are followed for project planning. Consistent, brief status reporting, issues management and meeting documentation are needed.
11. Employ the project plan as a tool for monitoring and managing the project. The plan should be used for:
 - Identification of tasks (WBS), determining of and planning for resources (staff, funds, etc.), and development of dependencies of tasks and resources.
 - Determination of achievable timetables and due dates.
 - Administration (status determination and reporting) of the project.
 - Selection of timeframes and scopes for quality assurance and technical architecture reviews.
 - Measurement of performance of vendors against schedule and deliverable commitments.
12. Watch for and manage "scope creep" on projects. Increasing functions or features of software or extending project requirements tend to raise costs, lengthen timetables and increase risks.

Lessons Learned from IT Projects

13. Watch for “creative development” on projects. Be careful of new, innovative, or unnecessary code or development techniques that will “improve deliverables”.
14. Assess the feasibility of time, money and personnel (skills and numbers) resources requirements for successfully completing the project. Ensure that all necessary resources (funds, staff, technology and other support and enabling services) are available and committed for assignment when and as long as needed.
15. Ensure adequate department internal staffing resources (numbers and skills) are committed to the project. Internal staffing must represent functional and technical areas, monitor compliance of vendors, and support the systems after implementation.
16. Develop realistic milestones and timetables. These should be derived from the project plans, using analysis tools that graphically display project tasks and schedules and determine critical paths, such as Gantt and PERT charts.
17. Assess and report project status (schedule performance, financial position, and key management and technical issues) on a frequent, periodic basis (at least monthly) to ensure vendors and other participating organizations are meeting quality, delivery and performance commitments. These include other departments affected by the project, vendors, subcontractors, oversight and regulatory bodies, and supporting and enabling organizations.
18. Involve all participants within and outside the ownership department in all planning and reporting activities for each project. These include other departments affected by the projects, vendors, subcontractors, oversight and regulatory bodies, and supporting and enabling organizations
19. Hold vendors and supporting and enabling organizations accountable for meeting milestones and performance commitments related to time, budget, deliverables and quality. Identify, research, explain and resolve deviations and take corrective actions in a timely manner. Deviations from the plan must be addressed promptly and decisively.
20. Clarify and formally document all project oversight roles (e.g., project management, steering committees, QA committees) to avoid overlap and duplication of efforts. Roles and responsibilities must be clearly defined and formally documented to avoid ambiguity.
21. Develop internal project policies, practices and procedures for identifying, researching, escalating, reporting and resolving issues, risks and other areas of importance to the success of the project. Project and risk management efforts must be supported by processes that facilitate the anticipation and identification of problems, enable them to be analyzed in a timely and effective manner, and allow for prompt and thorough follow-up for resolution.

Lessons Learned from IT Projects

22. Communicate relevant facts to all project participants and stakeholders on a frequent basis. Accurate, complete and frequent communication of project status and other pertinent information is an essential success factor for any project. Where possible, mass media capabilities (such as the “web” and e-mail) should be used to facilitate project communication.
23. Identify a formal notification channel. All parties must formally document a "notification" channel to be used in problem / change management issues and concerns. The formal notification channel must be used in all issue escalations, and the names, addresses, etc. must be kept up to date.
24. Develop and implement a life cycle support plan. Assign responsibility for development life cycle maintenance, and develop estimates of enhancement, maintenance, and operation cost for the projected life cycle of the application until retirement. As early as possible during the Conceptual and System Allocation phases of the software development life cycle, the approach for on-going enhancement and maintenance activities should be determined and projected costs should be estimated.
25. Evaluate and determine long-term system maintenance, operations, support costs and ensure adequate funding is available. Expenses for long-term support activities (e.g., software maintenance, system operations, communications, help desk, etc.) can be significant; therefore, they must be identified, realistically estimated, and budgeted early in the process.
26. Establish a formal configuration management plan and change control process for both the system development and maintenance / operations phases of the system development life cycle. Software configuration management involves establishing baselines and systematically controlling changes made to the baseline until application retirement.
27. Ensure that all approved deliverables are formally approved, baselined, and maintained under configuration management control. Changes to baselined deliverables must be formally logged, managed, and controlled.
28. Develop and implement basic project management processes designed to ensure that the system would be implemented within reasonable time frames, within budget, and meeting customer expectations. Settle on a system development life cycle that is appropriate for your project. Stay with the approach.
29. Develop and use good budget and time tracking tools. Resource tracking is key to achieving the budget commitment for the project. Ensure that the tools used to track expenses and work efforts assist in this project management tracking and oversight role. Use standard, proven tools and avoid redundant or duplicate reporting.

Lessons Learned from IT Projects

30. Ensure that agency management is strongly committed to the project. From top to bottom, agency management must demonstrate and re-enforce the organization's commitment to the project. Always gain senior level support for the project
31. Develop a formal schedule of team meetings and "one-on-one" sessions with the project staff. Use this as an opportunity to clarify the project status. Regular project status reporting is essential for project stakeholder information and understanding.
32. Ensure the stability of the resource pool. If possible, ensure that key project staff will remain with the project for the duration of development activities. Ensure that all project positions are appropriately "leveled" and that workload is balanced based on skill set and need. Good planning and defined process cannot make up for a lack of expertise and skills. Always choose the right team members.
33. Research with both public and private sectors on similar projects. Talk to both failed and successful projects, if possible. Get a view of the project based on broad-based first-hand experience.
34. Ensure that user training and documentation receive sufficient attention in the work breakdown structure (WBS). Most plans focus on deliverables and neglect user training and documentation until after implementation. Train early and often. Use training as a means to minimize the impact of change and facilitate organization change management.
35. Ensure that business functional requirements are prioritized. Manage business functional requirements to prevent scope creep and ensure that key requirements are delivered. Deliver high priority requirements. If any business functional requirements will not be delivered, or will be delayed, make sure they are low priority requirements.
36. Ensure that the project driver – time, budget, or functionality – is clearly defined and understood by all project stakeholders. Understand that you may have to adjust two of the factors to achieve the third (e.g., both time and budget may have to be re-baselined to achieve 100% of the business functional requirements).
37. Clearly defined and testable requirements are the foundation for project success. Unless the desired objectives are known, the expected results can not be achieved.
38. Ensure that end users are actively involved in the software development process - especially the design, testing, and acceptance processes. The software development process should be documented and easily understood by the stakeholders. Active participation by the user community is a key ingredient of project success.

Lessons Learned from IT Projects

39. Estimate at the task level. Project estimates should be the summation of detailed task estimates. Ensure that all stakeholders participate, from beginning to end, in the estimating process to ensure “buy in”. Use multiple estimating approaches to verify and validate estimates.
40. Ensure that the business functional requirements are verified and validated.
Requirements must be complete, accurate, and testable:
- The project manager must communicate the importance of creating a clearly documented set of requirements to all project stakeholders,
 - The project team must commit to understanding stakeholders’ business processes in stakeholders’ terms,
 - Project stakeholders must be actively involved in refining business functional requirements throughout the software development life cycle,
 - Requirements management software may be needed to manage a large set of business functional requirements or projects with very dynamic requirements’ changes,
 - Keep requirements’ changes to a minimum – use short iterations and prototyping to get feedback early.
41. Ensure that disaster recovery and business continuity planning are included in the project plan. Contingency plans must be verified and validated. Resources, both funding and time must be built into the plan. It is very important to **TEST** the plan.
42. Ensure that all required funding is identified and reserved prior to beginning the project. Failure to secure comprehensive project funding may result in project cancellation or reduced functional delivery.
43. Ensure that a solid permanent MIS staff with a common vision is on hand with the project before introducing contractual staff. Contractual staff should supplement permanent staff with an identified vision.
44. Ensure that the project team is united and the correct blend of skills exists on the team.
45. Ensure, through a formal process, that project team members make management aware of the potential problems at the earliest possible time and not spend an inordinate amount of time trying to resolve problems or determine who owns the problem. Small project delays add up to the point where schedule slippage cannot be recovered. Each problem needs to be resolved immediately to keep the project on schedule.
46. Require for large projects that both the integrator and the software vendor be available onsite each and every day for a full two weeks after implementation.

Lessons Learned from IT Projects

47. Develop a Service Level Agreement (SLA) with other agencies or groups to provide consistent service in a predictable manner and to allow the project teams to meet stated goals.
48. Schedule project activities that are delay prone earlier in the project. Any activity that is prone to delay from outside influences should be addressed as early in the project as possible in order to utilize built-in lag time in the overall project. This would include activities that involve internal and external auxiliary resources that are not under the direct control of the project team.
49. Be aware of and prepare project team to deal with special issues and high risks for large, long-term projects requiring approval from multiple federal agencies. Because of the federal regulations, projects are still pushed toward a large project, big bang implementation approach. To obtain federal matching dollars and approval, the federal process requires a planning effort for the whole initiative, which then requires an implementation plan of the whole initiative. It can become difficult to exit the planning phase of the project.
50. Spend more time listening to client's business needs and **understanding** the business before developing begins.
51. Ensure that a full-time project manager is assigned to the project. Key projects need full-time attention from a project manager dedicated to the system implementation effort.
52. Employ a "regional team" concept. Regional teams may be deployed to facilitate the implementation of statewide applications.

Lessons Learned from IT Projects

53. Ensure that team members are isolated from the day-to-day operations of the Department. Move team members to a separate office area and include the appropriate support tools (e.g., conference and meeting room areas) at the location.
54. Replan frequently. The project plan must be adapted to change and unforeseen circumstances. Ensure that the project plan is always current.
55. Hire subject matter experts (SME) early in the planning phase of the project. Hiring early in the process may prevent unplanned changes during the development phase. Hiring experts for a short engagement in specialized skill areas may facilitate the planning and estimating process.
56. Allow extra time for the procurement process. Estimates for hosting services, COTS procurement, and vendor consulting must be included in the project estimates with realistic target dates and deliverables.
57. Identify and define the roles, responsibilities, duties, and lines of authority of the responsible oversight and review bodies, including the extent and types of involvement respective to the products and services employed and the types of projects undertaken.
58. Make use of supplemental staff to augment permanent staff. Supplemental staff may be used to support post-implementation project activities since they have helped develop the system therefore they would be better able to troubleshoot application and data errors.
59. Consider more use of QA and testing resources in the early stages of the project.
60. Do not tie up all the investments and resources in a long-term vision for systems implementation. Much can be accomplished in the interim, while continuing to work towards an Enterprise solution. Where feasible and cost-effective, consider and implement interim solutions as you plan and execute your long-term solution.
61. Ensure the IT PM conducts integrative management across all of the affected project teams. In addition to the deliverables documents, technical interchange direction and outcomes and assigned action items were formally documented and agreed to across the project teams, contributing to integrative management.
62. Tap the expertise of very experienced customer resources. Ensure that their knowledgebase (in relation to engineering business requirements and business rules) is documented and confirmed by the sponsoring organization to represent true business needs.

Lessons Learned from IT Projects

63. Reinforce two-way communications by requiring PMs to document outcomes, issues, and action items of technical interchange and program management meetings.
64. Allow additional time in schedule for project approval, review, and sign off steps that are necessary for the process to be completed. The steps require duration rather than effort man-hours.
65. Identify a project single point of contact with the business users. It will help ensure accurate and timely communication, eliminate indecision, and maintain project scope and timeline.
66. Remember, in problem resolution situations where you are fixing a flawed system (triage), **prioritize** the fixes – fix those requirements that have the biggest impact to the users and ones that provide the basic needed functionality. Fix other problems later.
67. Provide frequent and comprehensive **training** for users of new business processes. Train the “call center” personnel on how to manage user questions and problems.
68. Never cut corners on testing. If you need more time, ask for it.
69. Attempt to find bottlenecks – areas that are slowing down the project. Resist the urge to hire more people until you understand what is slowing the project down.
70. Subject Matter Experts – Plan ahead. Ensure that business process subject matter experts are available at the appropriate times during the development process.
71. Consider having a back-up Project Manager available to address unexpected personnel changes.
72. Keep management informed and schedule meetings far in advance to ensure attendance of all required personnel.
73. Be aware of and be prepared to deal with risks associated with overdependence on contract personnel that can severely impact a project.

Lessons Learned from IT Projects

74. Ensure that technical documentation are reviewed prior to establishing budget and time lines.
75. Utilize a separate model (template) for estimating new projects versus estimating an existing system upgrade.
76. Publish /communicate upcoming projects to the entire agency to facilitate identification of interactions and/or needs for integration.
77. Set policy and guidelines for future projects with respect to source code ownership and security assignment.
78. Involve the clients from user agencies to aid / facilitate smooth transition and implementation of the project.
79. Include all organizations in planning before finalizing the project schedule.
80. Design, implement, and test the security model early, before development so that the environment is protected and appropriate accounts are used for their appropriate tasks.
81. Ensure that the customer is fully aware of the implications of scope changes and make sure that expectations are reset via constant communications.

Process and Technical Items

1. Clarify, determine and document the purchasing processes for IT products and services by accomplishing the following tasks:
 - Identify and define the roles, responsibilities, duties and lines of authority of the responsible oversight and review bodies, including the extent and types of involvement respective to the products or services employed and the types of projects undertaken. These organizations may include the ITS, Attorney General's office, Department of Administration (P&C), and other departments, commissions and boards having oversight, regulatory or support functions.
 - Research and develop appropriate and comprehensive procedures, policies, and/or checklists to guide and direct purchasing, contract administration and project management activities.

Lessons Learned from IT Projects

- Research and develop appropriate procedures, policies, and/or checklists to guide and direct enabling and supporting organizations (e.g., ITS) to create, price and deliver products and services for each project.
 - Develop standards, models and/or prototypes of RFPs, RFIs and contracts for reference by departments undergoing the IT purchasing and project implementation processes.
 - Identify the individuals and organizations responsible for the creation, approval and administration of state contracts.
2. Formalize and clarify the agency's software development process. Historically, RFP's have referenced IEEE standards and related integral processes for guiding software development activities. However, these need to be endorsed by the agency and enforced for all internally and externally developed software development projects.
 3. Conduct up front research on what's done elsewhere, that is, best of breed. Use of Meta and Gartner is an example.
 4. Investigate and resolve the issue of project management responsibilities in situations for which the ownership department may not possess the skills, experience and expertise to implement systems successfully. Accountability must be formally defined and accepted by both parties.
 5. Allow adequate time to complete satisfactorily the purchasing process, including RFP preparation, oversight and regulatory reviews, and vendor selection and contracting efforts. Ensure that the purchasing process is expedited. Ensure that the purchasing process is efficient and effective.
 6. Evaluate, before purchase, system components (including hardware, software and documentation) for acceptable performance, appropriate design, and expected quality, as well as conformance to state standards and specifications. Systems and system components must comply with the statewide technical architecture, and they must take advantage of or incorporate statewide technical initiatives. The intent is to ensure the designs and construction of systems are solid and documentation is complete, so they will meet performance requirements and can be enhanced, expanded and maintained easily and economically. Ensure that technical and business/program benchmarks are identified and employed in the process.
 7. Research, document and approve, where possible, required modifications to purchased software before purchasing. Determine in advance:
 - (a) Exactly what is being purchased,
 - (b) What will be changed, enhanced, or modified,

Lessons Learned from IT Projects

- (c) What are the processes and procedures for identifying, approving, documenting, designing, coding, testing, and accepting the proposed changes, and
 - (d) Who will support the product changes?
8. Evaluate, before purchase/contracting, the existence and adequacy of the vendors' quality (management and technical) practices, processes, and standards. Also, ensure that these are being followed in actual practice. Vendor procedures must be equal to or better than IEEE Standards.
 9. Ensure vendors have sufficient fiscal reserves/stability, staffing resources (numbers and skills), management acumen, and relevant experiences to accomplish each project successfully. Investigate and verify the financial strength, long-term business viability, and integrity of vendors through financial statements, credit checks (e.g., Dunn and Bradstreet), business references, and other available and appropriate means. Relevant financial and business credit statements should be included as a mandatory part of responses to RFPs.
 10. Investigate thoroughly the performance of vendors in similar types of engagements to ensure their capabilities to complete the work successfully. This can be accomplished by thoroughly checking references and completing other investigations.
 11. Employ the concepts of "best value" (versus "low cost") procurement / purchases to obtain maximum long-term benefits and performance enhancements. Areas requiring investigation and consideration include:
 - Describing business requirements versus detailed technical specifications (use the state's technical architecture as the underlying technical framework).
 - Employing "strategic partnerships" featuring longer-term and mutual gain agreements versus the traditional shorter-term and adversary state/vendor relationships.
 - Evaluating total life cycle costs and benefits versus strictly "low costs" decisions.
 12. Formalize the process for collecting and maintaining a repository of public agency LANs and WAN connections, and use GIS for reporting the physical locations of the connections. This information, particularly the GIS data, is essential for sharing connections and collaborating on IT resources (economies of scale) to reduce costs and increase the quality of services.
 13. Include all public agencies in the state's APMS and Federated Meta Data systems to attain a comprehensive repository of application systems and IT assets. This information is necessary for interagency and intergovernmental cooperation, as well as local and statewide planning for integrating and sharing IT resources to minimize costs and improve services.

Lessons Learned from IT Projects

14. Develop statewide communication security policies that incorporate all public agencies. This is necessary for achieving interagency and intergovernmental connectivity and interoperations in a secure, safe and reliable manner.
15. Work with the Office of Information Technology Services / Enterprise Technology Strategies (ITS/ETS) to update the statewide technical architecture for new and revised standards in as timely manner as possible.
16. Re-engineer business processes BEFORE implementing application systems. The automation, or re-automation, of present business/program processes gains few benefits. The intention is to use technology for allowing business processes to be consolidated and streamlined, so they become more efficient and effective; thereby, providing better and more responsive services at lower costs.
17. Evaluate commercial off-the-shelf (COTS) systems PRIOR TO selecting a custom development approach. In general, a COTS alternative is a cheaper and quicker approach for implementing applications; therefore, it provides greater benefits sooner at less cost.
18. Consider changing business processes to meet software designs rather than excessively customizing purchased software. It is often cheaper and easier to modify business/program processes to comply with the design of purchased software than to excessively modify COTS packages. Modifications take time and are expensive, and they create the potential for not being able to take advantage of future releases/upgrades of the package offered by the vendor. If COTS packages are modified, the changes should be kept to a minimum, and they should be isolated from the native COTS code to enable the applications to incorporate easily future releases of the software from the vendor.
19. Ensure that adequate product support is available and committed for all hardware and software, especially new technologies. Establish as much on-site, dedicated support as possible, especially for new products. This should be in the form of vendor personnel or vendor-trained personnel.
20. Ensure, through a formal process, that the entire organization is ready for the challenges and unknowns associated with new technologies. Ensure that all project staff are well educated on all application products. Constantly look to leverage any architecture / infrastructure support that is available. Train, and make available to the project team, a dedicated infrastructure resource to help set up the application architecture. Have a formal organization change management plan. Prepare the organization for the “culture shock” resulting from change.
21. Ensure, through a formal process, that a consistent approach is used to deploy applications and technologies across the agency. Develop and use for processes to

Lessons Learned from IT Projects

ensure that the application architecture conforms to both agency and ITS/ETS requirements.

22. Ensure that all application architectures are able to interconnect with State of North Carolina technical infrastructure. Facilitate application development by sharing reusable infrastructure components across the agency and throughout the project portfolio.
23. Ensure that agency partners are involved in architectural decisions. The ITS/ETS staff must ensure that major agency partners, especially the originating agency, are involved in discussions related to architectural decisions. Ensure that the agency has an approved Agency Technical Architecture that complies with the Statewide Technical Architecture. All agency projects must comply with the Agency Technical Architecture.
24. Where possible, use a “pilot” / “early adopter” approach to ensure that the proposed technology solution is both technically and financially feasible (proof of concept). The pilot site should be representative of the planned rollout sites and should be used to “beta” test the proposed solution. Pilot site results should be used to build the statewide rollout plan and should be used as primary input to the change management process.
25. Where possible, use a phased rollout approach for all projects requiring statewide implementation. The “big bang” approach to statewide implementation is too risky and difficult to support. Phased implementation addresses risk in a controlled manner and minimizes the impact of project defects on statewide services.
26. Ensure that conversion testing is built into the project plan. Conversion of legacy data and historical data is an onerous task that must be accomplished if the project is replacing an existing system. Be pro-active in testing and converting legacy system data.
27. Ensure that sufficient time is allocated for creating comprehensive analysis and design documents. Map project deliverables back to the design and analysis documents. Map design and analysis documents back to business functional requirements.
28. Ensure the transfer of skills from the development group to the operations and support group prior to system implementation. Use mentoring to facilitate skills’ transfer. Vendor-to-state staff skills’ transfer is critical if the vendor is only involved in development. Remember that the developer may not be on-site 90 days after project implementation.
29. Ensure that a dedicated test team is assigned and carries out thorough testing of each phase of the application prior to rollout.

Lessons Learned from IT Projects

30. Ensure that an appropriate and comprehensive Problem Report form that includes all the pertinent information is developed and used properly to report testing problems. A revision of the Problem Report form to ensure complete and accurate reporting of problems would result in less rework and quicker problem resolution.
31. Ensure, through a formal process, that architectural problems are caught and removed in the beginning of the development process.
32. Ensure that the database design is appropriate to support the technology and architecture of the application.
33. Ensure that application is broken into appropriate and manageable development “chunks”.
34. Ensure that the agency is involved and prepared with the technical architecture requirements, instead of depending on the vendor to research the requirements. In doing so, schedule slippage should be reduced.
35. Ensure that the project team coordinates activities with county and local communities. The project team should work with county and municipal governments to craft “community data sharing agreements” that will help in refreshing the database and ensuring accurate data.
36. Utilize secure internet-based applications. High availability, secure, internet-based applications should be implemented to facilitate citizen business transactions.
37. Allow time for parallel processing. Parallel processing gives the end users additional opportunity to become familiar with the new product between formal training and data conversion or “go live” activities.
38. Ensure that vendors and solution providers are recommending “state of the art” technologies. All technology solutions should be complicate with the Statewide Technical Architecture. Do not use old, outdated, or legacy technologies in a new application.
39. Utilize the use of a dedicated testing coordinator as a foundation for the parallel testing phase. The testing coordinator would develop formalized processes for testing and create other conceptual testing procedures that could aide in the ultimate development of a successful parallel testing phase.
40. Where possible, establish a prototype environment for early review of deliverables and code that could be reused / streamlined the development process.
41. Ensure that a formalized training and enrollment processes are established before offering a new enterprise service.

Lessons Learned from IT Projects

42. Allow additional time for more prototyping during certain stages of detail design.
43. Formalize the process for change requests, review and signoff.
44. Perform all initial loads and all delta data loads in the quality assurance environment, regardless of amount of data that is involved. It may extend the duration of the project, but load issues need to be found in the quality assurance client rather than in the production client.
45. Ensure that all existing equipment is in condition for connectivity to the new hardware and software. It reduces the time required to troubleshoot existing equipment in order to assure that the new installation is performing properly.
46. Ensure that Firewall configuration and testing activities are planned and performed as early as possible so that the customer site is up and productive “just-in-time”.
47. Ensure that the mount points have proper rights to allow installation and configuration of the COTS application (e.g., SAS) environment.
48. Be aware that data management is one the most difficult areas in decision support. Plan for contingency in terms of security, data quality, and performing tuning of the data management processes.
49. Install all services as automatic startup with appropriate dependencies in order to improve application availability.
50. Ensure that the clients play an active role in developing the test plan.

Lessons Learned from IT Projects

E-Government / Enterprise Approach Items

1. Design e-government applications well. They must possess a sound, logical and workable applications architecture and the code must be as error-free as possible. To accommodate expected loads and extended service windows, while satisfying the public's expectations for responsiveness, accuracy, and privacy and confidentiality; e-government applications must be extremely reliable, durable and secure.
2. Develop enterprise application design and coding standards. E-government applications must possess the toughness and resiliency necessary to handle the multitude of contingencies presented by the public, and design and coding errors must be minimized, so they will not adversely impact the performance of other applications exchanging information with them or sharing the same common services. Using common design and coding standards increases flexibility and reduces the probability of errors.
3. Develop and use enterprise standards, templates and best practices (including models for handling common business processing tasks). After the recommended and proven methods, techniques and models are created and documented, agency and contractor personnel should be trained (digital academy concept) to use them. The creation of these more detailed design and coding standards, approaches, and routines (possibly including recommended tools) will extend the statewide technical architecture to lower levels than presently presented.
4. Employ the concept of code reuse for common business functions. Best practices, standards and models may assist in developing quality designs and code; however, the inventory and reuse of these among like applications will eliminate duplications. This is especially true for code modules that perform common business tasks/steps (such a validating name and address).
5. Develop application code that is reusable for multiple applications. The practice of reusing proven application designs and code will improve the reliability of processing and shorten development and testing efforts, thus reducing cycle times and minimizing implementation and operation costs (particularly expenses for duplicating design and coding efforts and correcting errors from unproven components).
6. Document, inventory, and store in a central repository all 'best-of-breed' designs and modules. Improvements on a common design or code module stored in the repository would be designed, coded and tested once, and the latest version would be available to all applications using it. Likewise, new applications would start with proven designs and modify them to meet truly unique requirements. All applications would employ, to the greatest possible extent, common code modules available in the repository. Therefore, design and coding efforts for new applications would involve only the areas for which previously developed components were not available.

Lessons Learned from IT Projects

7. Use automated testing tools for each application implementation effort. Apply reuse concepts to tools, such as testing, that can be purchased once and used by all agencies. Using testing as an example, the large concurrent user load presented by e-government applications necessitates a more robust testing environment than normally required for traditional applications. Special tools are needed to meet this situation. Through the leveraging of the particular tool sets and personnel skills necessary to meet these more demanding testing needs, the state can save time and expenses by sharing these resources, instead of duplicating them.

Lessons Learned from IT Projects

8. Review project management practices for e-government applications.

The table below highlights some key technical differences between the two types of projects:

Type of System Development Project		
Technical Area	Traditional	E-Government
Requirements gathering	Rigorous and comprehensive	Limited due to reduced scope of application
Technical specifications	Extensive and robust	Limited due to use of common shared services and scope of application - assuming a proven technical framework or applications architecture is used
Testing and QA	Focus on quality	Focus on risk control
Risk management	Explicit, focused on major areas	Implicit, impacting all areas
Release process for ongoing improvements	Rigorous for less frequent and much bigger releases	Managed, but expedited for more frequent and smaller releases
User feedback	May be delayed and often must be solicited	Automatic and continuous from public interaction

Lessons Learned from IT Projects

9. Exercise good project management techniques. They are as important in the e-world as they are for traditional system development efforts. However, different emphases and approaches may be required. Major areas to reevaluate from traditional practices in the project management for e-government applications are schedule management, scope management, quality management and release management. Considerations for each are presented in the table below:

Area of Project Management	Considerations for E-Government Applications
Schedule management	<ul style="list-style-type: none">• Plan ahead for multiple releases - strive for regular release cycles• Make planning time and effort for each release consistent with the scope of release (planning should be 10-20 percent of total effort)
Scope management	<ul style="list-style-type: none">• Manage scope creep for each release• Priority sequence functions for planned release cycles
Quality management	<ul style="list-style-type: none">• Make risk-based decisions to eliminate <u>most serious</u> faults - technology allows prompt corrections to minor ones identified by the public after release is implemented
Release management	<ul style="list-style-type: none">• Classify releases on degree of complexity, scope and risk and manage accordingly - process and discipline are need for all types of releases, the amount varies by the complexity, scope and risk involved

10. Focus on the initial release of the e-government application. The first release of an e-government application involves more rigor, discipline and process - it contains many unknowns, and it is usually the largest and most involved. Other factors affecting the streamlining of approaches for project management include the availability of automated tools for testing and site integrity, the use of standards and proven designs, and the experience with and understanding of the business and technical environments by the project team. If these are present, quality assurance and project management processes can be streamlined and adjusted in a risk-based manner to become more cost-effective without impairing the success of the project.

Lessons Learned from IT Projects

11. Understand the differences between e-government applications and traditional applications. These differences are highlighted in the following table:

<u>Application Type</u>		
<u>Area</u>	<u>Traditional</u>	<u>E-Government</u>
Number of users	Few to hundreds of state staff	Thousands of citizens or businesses
Operating schedule	Eight to five - Monday through Friday	24X365 - with peak times on weekends and early evenings
Ease of use	May be complex, because only a few, well-trained personnel will use	Intuitive due to the wide diversity of public users
Responsiveness and accuracy of processing	Some ambiguity and fluctuations in responses can be tolerated by trained state staff	Public will not tolerate slow or inconsistent responses or errors in processing
Consistency of processing	May be inconsistent among applications, because few state staff use multiple ones	Public expects consistent look, feel and processing steps among agencies and applications
Use of common external technical services	Few - primarily middleware to exchange information among applications	Many - middleware to link new front-end to legacy processing and data bases, as well as employment of shared e-government services (e.g., credit card, e-forms, security, etc.)
Role of security to protect privacy and confidentiality and integrity of processing	Limited - application access restricted to only authorized personnel	Paramount - universal access to application through the Internet
Rate of change of application after implementation	Slow and paced - cycle times of releases in quarters or years	Varies but can be continuous - often with cycle times of releases in days or weeks

Lessons Learned from IT Projects

12. Ensure that ITS serves as a clearinghouse for technical lessons learned to include:

- Developing common application code that is reusable for multiple applications,
- Purchasing automated testing tools that may be used by all state agencies, and
- Providing written guidelines, templates, technical assistance, and training to reduce agency development time and present a more predictable project management environment.

13. Ensure that the organization is not new and the business infrastructure has had time to solidify before building an enterprise project.